

Lightning Functional Specification

Will Tuthill
Dan Aronson
Brewster Kahle
WAIS Inc
DRAFT 1.0
11/20/94

Lightning Functional Specification	3
Component Overview	3
Rainman Feed Handler	4
WAIS DBMS	4
WAIS Filer Library	4
WAIS Search Library	5
HTML Generator	5
Design Overviews	6
Rainman Feed Handler	6
WAIS DBMS	25
WAIS Filer Library	26
WAIS Search Library	27
HTML Generator	28

This specification is in response to: Thunder and Lightning Taking the Internet by Storm with AOL Publishing Tools by Sunil Paul and William Marriott, May 25, 1994, where the initial project was defined as:

Lightning -- A fast-track technology that lets any AOL IP publish to the Internet with little effort. The Lightning strategy is to position AOL's Rainman Plus as a platform from which to publish to Internet formats. Just as word processing applications gain market share by complying with their competitor's formats, Lightning will make Rainman Plus a more attractive interactive publishing platform. The target delivery of the product is September/October 1994. (See the attached screen shots of Mosaic and Rainman Plus for Time, Inc.)

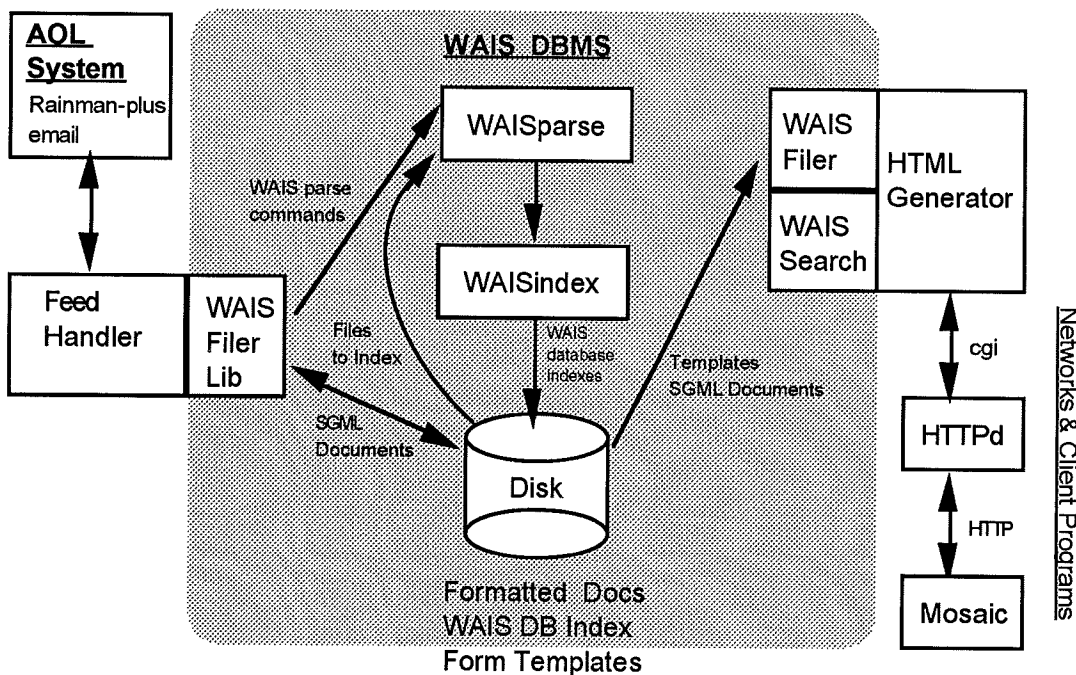
Features of the System:

- Integrated system for IP's to publish on Internet using Rainman system.
- Graphics and text integrated into a form will be displayed in a similar format HTML
- List boxes will be translated to a list of actionable hypertext buttons in HTML.
- Database searches will translated to form-based HTML entry boxes that can then be tied to WAIS searches.
- Information Providers will have the ability to select on a form by form level the ability to translate to HTML for Internet publication.
- There will be no provision for sales and credit-card transactions over the Internet with this translation.
- Text in a scroll box will be presented as a separate HTML page with the first x characters visible as a clickable hypertext button.
- Chat, Newsgroup/MsgBrds, software libraries, non English documents, and Surveys will not be supported.

This functional specification outlines the components and operation of the Lightning system. This draft concentrates on the RAINMAN input and document filing. It is not a full specification of the system.

This system duplicates the AOL Information Provider services in an Internet service using HTTP and HTML (World Wide Web). This system does not handle discussion boards, message boards, conferencing or ties to other AOL services. In other words, this will build a similar service to the broadcast-oriented publishing services of AOL.

Since America Online personnel understand many of the implications of implementing Rainman-plus, we suggest these are our capabilities, but would need further work with the experts to find difficult areas.



Lightning Block Diagram

Component Overview

This section provides an overview of the technical components of the Lightning system. A synopsis of each component is expressed in the following form:

- Functions:** Summarizes the tasks handled by the component.
- Inputs:** Describes the data entering the component and where it originates.
- Outputs:** Describes the data produced by the component and where it goes.
- State:** Specifies any internal state for the component.

Rainman Feed Handler

- Functions:** Translate incoming Rainman email messages to WAIS Filer Library calls to create, modify, delete and retrieve documents.
- Inputs:** Rainman+ email messages from AOL.
- Outputs:** Function calls to WAISfiler library to create, delete, get and modify documents.
- Reply email to sender of Rainman script.
- Record of each transaction to a log file.
- State:** Internal parser state for processing Rainman blocks.

WAIS DBMS

- Functions:** Index documents for rapid search and retrieval.
- Inputs:** Documents to index for searching via waisparse and waisindex.
- Outputs:** WAIS index files on disk.
- State:** Stored document indexes.

WAIS Filer Library

- Functions:** Translate document specifications to SGML formatted files on disk.
- Update SGML formatted files with document modifications.
- Retrieve and delete SGML formatted documents.
- Inputs:** Function calls from the Feed Handler to create, delete, retrieve and modify documents.
- Formatted files on disk for input to document retrieval and modification functions.
- Outputs:** Document ID requests to WAISserver.
- Formatted SGML document files on disk.

State: None.

WAIS Search Library

Functions: Serve document references to HTML generator.

Inputs: Search requests from HTML generator.

Outputs: File IDs and headlines to satisfy search requests from the
HTML generator.

State: None.

HTML Generator

Functions: Use form templates and SGML documents to create Mosaic-ready
HTML documents.

Inputs: Document display request from HTTPd.

Documents from WAIS DBMS.

Templates from the form template database.

Outputs: HTML document to display via HTML viewer.

State: None.

Design Overviews

This section provides an overview of the design of each component of the Lightning system. It provides a more in-depth descriptions than the Component Overview including data structures and data transfer mechanisms used between components.

Rainman Feed Handler

All of the Rainman specific code will be in the feed handler. Therefore this module should be carefully designed and built with AOL involved.

Functionality

The Rainman Feed Handler translates incoming Rainman email messages into WAIS Filer Library calls which result in adding, removing, modifying and retrieving documents in the WAIS database. Email messages are assumed to be sent directly from AOL and are assumed to contain valid, error-free Rainman scripts. The email header information is removed by the feed handler and the Rainman script is parsed. The results of parsing a Rainman script is a series of library calls to the document filer library. There are 4 library calls which facilitate the addition, deletion, modification and retrieval of documents in the WAIS database. Each library call requires information about the document such as its Rainman ID, the current user and the data contained within the document. This data is accumulated in the appropriate data structures as the Rainman script is parsed. Each time a valid block of Rainman code is parsed, a library call to create, delete or modify a document is made to the WAIS Filer Library.

Following is a summary of which commands will be supported in version 1.0. Y means supported. N means not supported. P means partially supported.

Command	Supported

ATTACHFILE	N
CLEARDATA	Y
COMMENT	Y
CONTEXT	Y
CREATE	Y
DATA	Y (only formats TEXT, GIF)

DATELINE	Y
DELIM	Y
ECHO	Y
END	Y
ENDTRY	N
ERRORS	P (not full error reporting)
FBLANGUAGE	N
FEEDBACK	N
FIELDTEXT	Y
FORM	Y
FORMAT	P
GROUP	Y
HEADLINE	Y
HTEXT	Y
ICON	Y
ID	Y
INSERT	Y
KILL	Y
LANGUAGE	P (only English)
MAP	N
MAPKEY	N
MAXSIZE	Y
MODIFY	Y
OPTIONS	Y
PASSWORD	N

PLUSGROUP	N
POSITION IN	Y
REFERENCE	N
RELEASE	Y
REMOVE	Y
SEARCHWORDS	Y
SLUG	N
STOP	N
TEXT	Y
TEXTKEY	N
TITLE	P (only supports TEXT)
TRY	N
USER	Y
VERSION	N
VIEWRULE	N
WARNINGS	N

Data Structures

The information required to complete each addition, deletion, modification or retrieval of a document is accumulated in a list containing the following data structure:

```
ArgSpec {  
    name      /* Name of argument */  
    value     /* Value of argument */  
}
```


The valid names for arguments are described below along with their meanings and the format of their values.

WfNtitleText	The document title string,
WfNtitleDateSpec	How to display document creation date in the title expressed as a bitfield. Values in the bitfield are set by ORing together any combination of the following:
ACTIVE	Set to show date and/or time in title. The default is to not show it.
SHOW_DATE	Show creation date.
SHOW_TIME	Show creation time.
SHOW_ZONE	Show time zone abbreviation
SHOW_BEFORE	Show date and/or time before title. The default is to show it after the title.
WfNlinkTo	Pointers to other documents expressed as a list of LinkSpecs where a LinkSpec is defined as a list of ArgSpecs with argument names as follows:
WfNtitleText	The link title string.
WfNtitleDateSpec	How to display the creation date.
WfNiconNumber	Number of icon for link.
WfNviewRuleNumber	Number of view rule for link.
WfNreleaseTime	When link is released to users.
WfNremovalTime	When link is revoked from users.
WfNlinkToID	
WfNlinkFrom	List of Rainman IDs of documents (Collections) which point to this document.
WfNlinkLimit	Maximum number of links expressed as an integer.
WfNlinkOrder	Ordering rule(s) for links. TBD.

WfNlinkScrolling	Whether links scroll out of a document expressed as a boolean.								
WfNformNumber	Number of form to use in displaying the document expressed as an integer.								
WfNiconNumber	Number of icon used to represent the document expressed as an integer.								
WfNiconShowing	Whether or not to show icons expressed as a boolean.								
WfNdatabase	Database in which document is stored expressed in a form which is TBD.								
WfNdataSpecs	A list of document DataSpecs where a DataSpec is defined as a list of ArgSpecs with argument names as follows: <table data-bbox="526 814 1214 1060"> <tr> <td>WfNfieldNumber</td><td>Field in which data appears.</td></tr> <tr> <td>WfNformat</td><td>Format of the data.</td></tr> <tr> <td>WfNbytes</td><td>Number of bytes of data.</td></tr> <tr> <td>WfNdata</td><td>Pointer to the data itself.</td></tr> </table>	WfNfieldNumber	Field in which data appears.	WfNformat	Format of the data.	WfNbytes	Number of bytes of data.	WfNdata	Pointer to the data itself.
WfNfieldNumber	Field in which data appears.								
WfNformat	Format of the data.								
WfNbytes	Number of bytes of data.								
WfNdata	Pointer to the data itself.								
WfNplusGroup	Plus group id for usage tracking expressed in a form which is TBD.								
WfNsearchWords	Additional words to attach to this document which are used in searching. These are expressed as a NULL separated list of strings terminated with a NULL.								
WfNtime	Time at which to perform a KILL request expressed in a form TBD.								

Inputs

After the email header is stripped off of the incoming Rainman script, the Rainman commands themselves are processed. As the parser processes Rainman commands, internal state variables are set and data describing the document being created or modified is accumulated in ArgSpec structures. The actions taken for each Rainman command are detailed below.

ATTACHFILE

This command is ignored.

CLEARDATA

Internal state (modify block)

Clears data from the specified field(s) of a given document.

Create and initialize DataSpec(s)

WfNfieldNumber = field number to delete

WfNformat = format of data (GIF, JPEG, ART)

WfNbytes = 0 to clear

WfNdata = NULL to clear

COMMENT

Lines beginning with the comment command are ignored.

CONTEXT

Internal state (modify block)

Specify operation upon a given link.

Set docLinkNumber = specified link

CREATE

Internal state (create block)

Start the beginning of a new create block by resetting document state variables. Trigger the end of previous create or modify block if no END command did. (See END for details.)

Only objects of type ARTICLE or COLLECTION are supported.

DATA

Internal state (create/modify block)

Specify data for a given field of the document.

Create and/or initialize a DataSpec:

WfNfieldNumber	= field number for data
WfNformat	= format of data (GIF, JPEG, ART)
WfNbytes	= length of data in bytes
WfNdata	= pointer to the data

DATABASE

Internal state (create/modify block)

Specify the database the document resides in.

Create and/or initialize an ArgSpec:

argSpec.name	= DtNdatabase
argSpec.value	= specified database (possibly NONE)

DATELINE

Internal state (context, create/modify, outside)

Specify if and how to display the document creation time in the document title.

Create and/or initialize a TitleDateSpec by using bitwise OR of any combination of the following:

ACTIVE
SHOW_DATE
SHOW_TIME

SHOW_ZONE

SHOW_BEFORE

DELIM

Interpreter only

Change the delimiter string used by the Feed Handler.

ECHO

Interpreter only

Write the specified string to the email buffer.

END

Internal state (create/modify block)

Signals the conclusion of a create or modify block. The information about the object collected in the data structures described in each command is processed at this point. Also processed at this point are requests to modify other documents to point to the currently specified document.

Make a function call to the document filer library to create or modify the document.

ENDTRY

Interpreter only

Terminate handling of try block and resume processing. (Not in prototype.)

ERRORS

Interpreter only

Sets error handling status to indicate if all errors are to be considered fatal. Since we're assuming the processing of RAINMAN scripts by AOL prior to interpreting them, we do not intend to implement sophisticated error handling. Rough feedback regarding errors will be provided to the user.

FEEDBACK

Interpreter only

Sets the feedback level to indicate which errors, warnings, etc. are provided to the user by adding them to the email buffer.

FIELDTEXT

Internal state (create/modify block)

Specify the text contained in a particular data field of the document.

Create and/or initialize a DataSpec

WfNfieldNumber	= field number for data
WfNformat	= text
WfNbytes	= length of text in bytes
WfNdata	= pointer to the text

FORM

Internal state (create/modify block)

Specify the form number to use when displaying the document.

Create and initialize an ArgSpec:

argSpec.name	= WfNformNumber
argSpec.value	= specified form number

FORMAT

Internal state (create/modify block)

Create and initialize a FormatSpec:

FormatSpec TBD

GROUP

Interpreter only (TBD)

Set the group id to indicate which group is being used.

HEADLINE

Internal state (create/modify block or link context)

Specify the headline for a document or link.

When referring to a DOCUMENT or ARTICLE

Create and initialize an ArgSpec:

argSpec.name = WfNtitleText

argSpec.value = text of headline

When referring to a link

WfNtitle = text of headline

HTEXT

Internal state (create/modify block)

Specify the text used in a particular field of the document and use the beginning of the same text for the headline.

Create and/or initialize ArgSpec & DataSpec:

argSpec.name = WfNtitleText

argSpec.value = first width - 3 characters of text

WfNfieldNumber = 0 TBD
WfNformat = text
WfNbytes = length of text in bytes
WfNdata = pointer to the text

ICON

Internal state (create/ modify block or link context)

Specify the icon which represents the document or link.

When referring to a COLLECTION or ARTICLE

Create and initialize ArgSpec:

argSpec.name = WfNiconNumber

argSpec.value = number of icon

When referring to a link

WfNiconNumber = number of icon

ID

Interpreter only

Call CREATE handler for an ARTICLE since ID is just an abbreviated command to create an article.

INSERT

Internal state (create/ modify block)

Add a link to the current document.

Create and initialize LinkSpec:

WfNnumber = specified number or getNextLinkNumber()

if docReleaseTime valid

WfNreleaseTime = docReleaseTime

else

WfNreleaseTime = defaultReleaseTime

if docRemovalTime valid

WfNremovalTime = docRemovalTime

else

WfNremovalTime = defaultRemovalTime

if docViewRule valid

WfNviewRule = docViewRule

else

WfNviewRule = None

KILL

Remove a document from the database.

Create and initialize ArgSpec

argSpec.name = WfNtime

argSpec.value = time specification (defaults to NOW)

Call the document delete function in the document filer library with the arg.

All links to the document are removed from the database at the specified time and the document itself is removed from the database one hour later.

LANGUAGE

Only English language documents are supported.

MAP

Not supported.

MAPKEY

Not supported.

MAXSIZE

Internal state (create/modify block or outside)

Specify the maximum number of links which can appear in a document.

When block is create or modify

Create and initialize ArgSpec:

argSpec.name = WfNlinkLimit

argSpec.value = specified maximum number of links

When outside a create or modify block

set defaultLinkLimit = as specified

MODIFY

Internal state (modify block)

Trigger end of previous create/modify block. (See END for details.) Start the beginning of a new modify block by resetting document state variables and retrieving the current state of the document from the WAIS database.

OPTIONS

Internal state (create/modify block)

Specify how links are inserted and displayed.

Create and initialize ArgSpec(s)

When SCROLL is specified

argSpec.name = WfNlinkScrolling

argSpec.value = true or false as specified

When MINI_ICONS is specified

argSpec.name = WfNshowIcons

argSpec.value = true or false as specified

When link order is specified

argSpec.name = WfNlinkOrder

argSpec.value = order type which may be one of the following:

NEW_TO_OLD

OLD_TO_NEW

ALPHABETIC

REVERSE_ALPHABETIC

ALPHANUMERIC

REVERSE_ALPHANUMERIC

PASSWORD

Not supported.

PLUSGROUP

Internal state (create/modify block, outside)

Create and/or initialize an ArgSpec:

argSpec.name = WfNplusGroup

argSpec.value = specified plus group

POSITION IN

Internal state (insert block in create/modify block)

Add a link to the specified document which points to the current document.

Create and/or add specified document ID in linkFrom list in current document

argSpec.name = WfNlinkFrom

argSpec.value = linkFrom

Modify document linked from - add link in linkTo list

create ArgSpecs for linkTo, linkLimit, linkOrder, linkScrolling

docGet(user, ID, args);

For normal positioning

insert link as specified by linkOrder in linkTo list

For absolute positioning

insert link into specified position in linkTo list

For relative positioning

insert link in specified relative position in linkTo list

TBD: HEADER, BODY, FOOTER positioning

REFERENCE

Not supported.

RELEASE

Internal state (create/modify, insert/context, outside)

Specify when to make link appear to the users.

When in insert or context block

WfNreleaseTime = specified release time

When in create or modify block

set docReleaseTime = specified release time

When outside any of these blocks

set defaultReleaseTime = specified release time

REMOVE

Internal state (create/modify, insert/context, outside)

Specify when to remove a link from user view.

When in insert or context block

WfNremovalTime = specified removal time

When in create or modify block

set documentRemoveTime

When outside any of these blocks

set defaultRemoveTime

SEARCHWORDS

Internal state (create/modify block)

Specify additional words to attach to the document which are used in a database search.

Create and/or initialize ArgSpec:

argSpec.name = WfNsearchWords

argSpec.value = NULL separated list of search words to add

SLUG

TBD. (Not in prototype.)

STOP

Not supported.

TEXT

Internal state (create / modify block)

Specify text of a document.

Create and / or initialize DataSpec(s)

WfNfieldNumber = 1

WfNformat = text

WfNbytes = length of text in bytes

WfNdata = pointer to the text

TEXTKEY

Not supported.

TITLE

Same as HEADLINE but will support typed (non-ASCII) text at some point. Only TEXT (not TYPED TEXT) is supported for the first release.

TRY

Interpreter only

Put marker on stack to indicate where to roll back to upon fatal error.

USER

Internal state

Set user = specified user

VERSION

Interpreter only

Set version flag = specified version (supports only 2.0)

VIEWRULE

Internal state (create/modify, insert/context blocks)

Specify a rule number which indicates who can see the given document or link.

When in create or modify block

set docViewRule = specified view rule

When in insert or context

WfNviewRule = specified view rule

WARNINGS

Interpreter only

Set flag to indicate level of warnings.

Outputs

The results of parsing the Rainman input is a series of function calls to the WAIS Filer Library which create, modify, delete and retrieve documents from the WAIS database. The interface to the library is as follows:

status wfCreate(ID, args)

Adds the specified document to the WAIS database.

status	wfDelete(ID, args)
	Removes the specified document and links to it from the WAIS database.
status	wfModify(ID, args)
	Set the specified characteristics of the specified document.
status	wfRetrieve(ID, args)
	Get the specified characteristics of the specified document.

Email messages are returned to the originator with status, warnings, errors etc. Since this whole system is mirroring the AOL site, we will not be implementing all of the elaborate reporting mechanisms in AOL. On the other hand, it will be careful to not "crash and burn" on errors.

Each document creation, deletion and modification request is logged to a file.

State Variables

There are 2 sets of state variables, one which is reset upon the start of parsing a new Rainman script and one which is reset upon the start of parsing a new document. These state variables are described below.

These session-specific state variables are reset upon reading a new Rainman script.

Delimiter	String used to delimit rainman commands.
EmailBuffer	File used to store email reply to sender of script.
ErrorProcessing	Error processing state.
FeedbackLevel	Feedback level.
Plusgroup	Plusgroup for document.
Group	Group for document.
IpId	ID for information provider.
ReleaseTime	When to release link. Defaults to NOW.
RemovalTime	When to remove link. Defaults to NEVER.
Database	Database into which documents are to be stored.

LinkLimit	Maximum number of links a document may have.
------------------	--

These document-specific state variables are reset upon the start of processing a new Rainman document.

ReleaseTime	When to release link. Defaults to NOW.
RemovalTime	When to remove link. Defaults to NEVER.
LinkNumber	Next link number to use.
LinkViewRule	View rule number to use.
TitleDateSpec	If and how to display the creation date in the title.
LinkFromDocs	Pairs of IDs & modified linkTo lists.

Installation & Initialization

This module is put as a command in the aliases file of a unix machine. Therefore a separate process gets created for each message that comes in. This has the benefit of being able to handle a large number of messages simultaneously. It has the disadvantage that commands to the WAIS DBMS must be serialized.

Upon startup, the session-specific and document-specific state variables are initialized to their default values.

WAIS DBMS

Functionality

The WAIS DBMS holds the documents specified by Rainman commands and serves these documents to users via the HTML Generator.

Data Structures

There are no data structures used to interface with the WAIS DBMS.

Inputs

Document indexing commands in the form of WAISparse and WAISindex invocations.

Document requests from HTML Generator.

Outputs

WAIS indexes are created on the disk as a result of indexing requests on new or modified documents.

Documents are returned to HTML Generator.

State Variables

Documents are stored in the database. Information about existing documents is stored in the indexes.

Installation & Initialization

The WAIS DBMS is installed and initialized in the standard way as described in documentation available with the system.

WAIS Filer Library

Functionality

The role of the WAIS Filer Library is to provide a simple interface by which the Feed Handler can store and access documents. The library provides 4 function calls to create, delete, retrieve and modify documents in the WAIS database.

Data Structures

The data structures for this component are described in the outputs section for the Feed Handler.

Inputs

The function calls described in the outputs section of the Feed Handler are the inputs to the Document Filer Library. These commands enable the creation, deletion, modification and retrieval of documents from the WAIS DBMS.

In the case of a document retrieval request, documents in SGML format are read from the disk.

Outputs

Commands are created for indexing files in the WAIS DBMS using the WAISparse and WAISindex interface.

In the case of a call to the document creation or modification function, the new or modified document is stored on disk by the WAIS Filer Library. Documents are stored in SGML format.

State Variables

None.

Installation & Initialization

The WAIS Filer Library is linked with the Feed Handler and HTML Generator at compile time. The library may be linked statically or dynamically.

WAIS Search Library

Functionality

The role of the WAIS Search Library is to provide an interface through which the HTML Generator can make search and retrieval requests.

Data Structures

The data structures for this component are URL specifications for HTML.

Inputs

Search requests from the HTML generator in the form of queries using the HTML forms and cgi-bin facilities.

WAIS indexed documents on disk.

Outputs

Document IDs and headlines to satisfy search requests.

Document IDs to satisfy retrieval requests.

State Variables

None.

Installation & Initialization

The WAIS Search Library is linked with the HTML Generator at compile time. The library may be linked statically or dynamically.

HTML Generator

Functionality

Use form templates and SGML documents to create Mosaic-ready HTML documents. This includes providing a user interface to document search and retrieval.

Data Structures

There are no data structures used in the interface with the HTML Generator.

Inputs

Document display request from HTTPd through CGI (Common Gateway Interface.)

Indexed documents stored in WAIS DBMS.

HTML templates from the form template database.

Outputs

HTML documents which are displayed via an HTML viewer such as Mosaic when the HTML Generator is invoked through the CGI.

State Variables

Document IDs are stored in the clickable URLs in the HTML between search and retrieval requests.

Installation & Initialization

The HTML Generator is installed in the cgi-bin directory in a directory called Lightning.

Form 49

Main text (field 1; NY 12)

Transmitted: 94-08-10 11:32:54 EDT (demo_49)
FDO #40-3833

Graphic Field (field 5)

Dimensions 198 x 198



Caption (field 11)

form 57

Graphic Field (field 5)
Dimensions: 400 x 160



Main Text (field 1)
FDO #40-3844

Form 61

Graphic Field (field 5)

Dimensions 198 x 198

Headline (field 3; NY 14; 18 char)

Main Text (field 1; NY 12)

Transmitted: 94-08-10 10:21:27 EDT (demo_61)

FDO #40-3845

Caption (field 11)

Headline (field 3;NY 18;23 char)

Graphic Field (field 5)
Dimensions: 200 x 100



Caption 1 (field 11;NY 9;19 char)

Graphic Field (field 6)
Dimensions: 200 x 100



Caption 2 (field 12;NY 9;19 char)

Main text (field 1;NY 12)

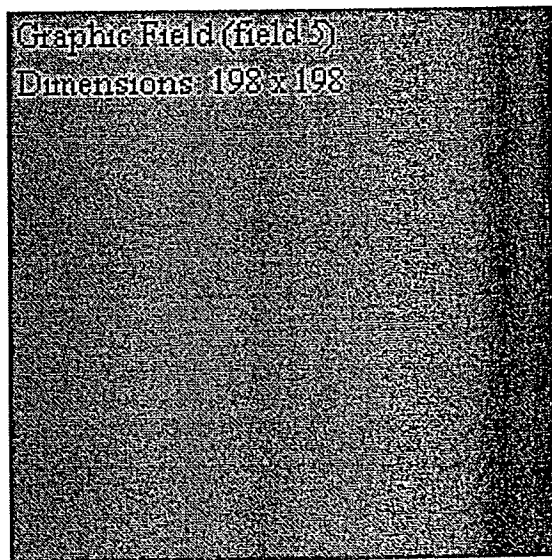
Transmitted: 94-08-10 11:28:55 EDT (jntest89)
FDO #40-4145

Headline (field 3; NY 18; 23 char)

Main text (field 1; NY 12)

Transmitted: 94-08-10 11:35:47 EDT (demo_113)
FDO #40-5021

Graphic Field (field 5)
Dimensions: 198 x 198



Caption (field 11; NY 9; 18 char)

Form 117

Headline (field 3; NY 18; 23 char)

Main Text (field 1) ·
FDO #40-5794

Form 121

Headline (field 3; NY 14; 23 char)

Graphic Field (field 5)

Dimensions: 400 x 130



Main Text (field 1)
FDO #40-5795

Form 125

Headline (field 3; NY 14; 23 char)

Graphic Field (field 5)
Dimensions: 235 x 200
FDO #40-5797

Graphic Field (field 6)
Dimensions: 235 x 200

Caption (field 11; NY 9; 21 char)

Caption (field 12; NY 9; 21 char)

Headline (field 3; NY 14; 28 char)

Graphic Field (field 5)
Dimensions: 200 x 250



Main Text (field 1)
FDO #40-5791

Form 133

Headline (field 3; NY 18; 21 char)

Graphic Field (field 5)

Dimensions: 475 x 210

FDO #40-5799